UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/683,336	12/14/2001	David S. Hardin	01H1533	8636

24234 7590 03/18/2005

SIMMONS, PERRINE, ALBRIGHT & ELLWOOD, P.L.C.
THIRD FLOOR TOWER PLACE
22 SOUTH LINN STREET
IOWA CITY, IA 52240

EXAMINER	
ZHEN, LI B	

ART UNIT	PAPER NUMBER
2126	

DATE MAILED: 03/18/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/683,336	HARDIN ET AL.	
	Examiner	Art Unit	
	Li B. Zhen	2126	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *05 March 2002*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-38* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-38* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1 – 38 are pending in the application.

Claim Rejections - 35 USC § 101

2. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

3. Claims 1 – 28 are rejected under 35 U.S.C. 101 because they are directed to non-statutory subject matter.

4. Claims 1 – 28 are directed to method steps which can be practiced mentally in conjunction with pen and paper, therefore they are directed to non-statutory subject matter. Specifically, as claimed, it is uncertain what performs each of the claimed method steps. Moreover, each of the claimed steps, inter alia, running, receiving, determining, servicing, activating, assigning, generating, monitoring, aborting, transferring, sending, holding, discerning, reserving and suspending, can be practiced mentally in conjunctions with pen and paper. The claimed steps do not define a machine or computer implemented process [see MPEP 2106]. Therefore, the claimed invention is directed to non-statutory subject matter. (The examiner suggests applicant to change "method" to "computer implemented method" in the preamble to overcome the outstanding 35 U.S.C. 101 rejection).

Claim Rejections - 35 USC § 112

5. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

6. Claims 3 – 21 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

7. Claim 3 refers to the method of claim 3. There is insufficient antecedent basis for this limitation in the claim.

Claim Rejections - 35 USC § 102

8. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

9. **Claims 1 – 38 are rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent No. 6,374,286 to Gee et al. [hereinafter Gee].**

10. As to claim 1, Gee teaches a method for managing interrupts [An 16-input priority interrupt controller 272; col. 10, lines 43 – 50] in a multiple virtual machine environment [three JVMs, designated as JVM0, JVM1 and JVM2, Fig. 12; col. 23, line 65 – col. 24, line 18 and col. 23, lines 18 – 29], comprising the steps of:

running concurrently a plurality of independent virtual machines [col. 23, line 65 – col. 24, line 18], each virtual machine having associated therewith a plurality of anticipated interrupt signal types [services any interrupts which have occurred and queues the next partition to assume processor control; col. 23, lines 30 – 55];

receiving a plurality of interrupt signals [inputs are captured in a storage register (not shown) and a mask (MASK) from the register file 204 determines which interrupts will be recognized; col. 10, lines 43 – 50];

determining which interrupt signal of the plurality of received interrupt signals has the highest priority [priority encoder (not shown) generates the identifying number of the highest-priority unmasked interrupt; col. 10, lines 43 – 50]; and

servicing the interrupt signal determined to have the highest priority [col. 21, lines 37 – 43; proxy thread handles JVM-specific interrupts, col. 3, line 57 – col. 4, line 3].

11. As to claim 29, Gee teaches an interrupt management system [col. 10, lines 43 – 50] for an apparatus capable of running multiple concurrent virtual machines [col. 23, line 65 – col. 24, line 18 and col. 23, lines 18 – 29], comprising:

a timer component comprising a plurality of virtual machine timers [partition interval timer 1712; col. 28, lines 45 – 52], said timer component further comprising an active virtual machine switch signal output [1712 is only loaded via a load register 1714 which specifies the time interval for the next partition time slice; col. 28, lines 52 – 67];

a multiple virtual machine control component [partitions are controlled by a "master" JVM which always operates in partition 0; col. 23, lines 19 – 29], comprising an

active virtual machine identification signal output [proxy thread then pushes the flags and JVM ID onto the stack and executes a resumeJVM instruction which starts the actual JVM thread for the partition; col. 23, lines 30 – 44];

a processor component [JEM processor 100, Fig. 1; col. 8, line 57 – col. 9, line 3], coupled with said timer component [col. 28, lines 45 – 52];

an interrupt controller component [col. 10, lines 43 – 50] coupled with said processor component and with said timer component, said interrupt controller component comprising an active virtual machine identification signal input [active partition ID is maintained by a single partition register 1746; col. 30, lines 55 – 67] coupled with said active virtual machine identification signal output [col. 11, lines 18 – 38], said interrupt controller component also comprising an interrupt signal input [16-input priority interrupt controller; col. 10, lines 43 – 50]; and

a memory component storing interrupt handler code [interrupt handler thread of control; col. 24, lines 37 – 43].

12. As to claim 35, Gee teaches an interrupt controller [col. 10, lines 43 – 50] for a multiple virtual machine environment [col. 23, line 65 – col. 24, line 18 and col. 23, lines 18 – 29], comprising:

an interrupt signal input [16-input priority interrupt controller; col. 10, lines 43 – 50];

a plurality of virtual interrupt latch components coupled with said interrupt signal input [data and address interface 262 and 242, address incrementers 236, instruction

register 266, parsing logic 270, the arithmetic-logic unit (ALU) 256, shifters 254, and a multiport register file 204; col. 10, lines 43 – 57]; and

a plurality of global interrupt mask registers [a mask to the interrupt controller; col. 10, line 57 – col. 11, lines 9];

wherein each global interrupt mask register of said plurality of global interrupt mask registers is coupled with one of the virtual interrupt latch components [col. 20, line – col. 21, line 5].

13. As to claim 38, Gee teaches a processor-based interrupt signal management system [col. 10, lines 43 – 50] for a multiple virtual machine environment [col. 23, line 65 – col. 24, line 18 and col. 23, lines 18 – 29], comprising:

an integrated circuit chip [JAVA embedded microprocessor (JEM); col. 8, lines 47 – 58], comprising;

a processor component [col. 8, line 57 – col. 9, line 3];

a multiple virtual machine management [col. 23, line 65 – col. 24, line 18 and col. 23, lines 18 – 29] component coupled with said processor component, said multiple virtual machine management component comprising a plurality of virtual machine activation timer components [partition interval timer 1712; col. 28, lines 45 – 52];

a memory component [memory 104, Fig. 1; col. 8, line 57 – col. 9, line 3] coupled with said processor component, said memory component comprising interrupt handler code [interrupt handler thread of control; col. 24, lines 37 – 43];

a memory access error input [memory access error; col. 31, lines 26 – 35];

an active virtual machine identification output [active partition ID is maintained by a single partition register 1746; col. 30, lines 55 – 67]; and

a memory access location output [FieldOffset field, along with the objectref pointer, identifies a location in memory of the field; col. 13, line 65 – col. 14, line 9]; and

an external memory protection component [PMU enforces memory protection constraints and contains the watchdog timers that enforce the context switches that change the partitions; col. 27, lines 23 – 30], not located on said integrated circuit chip, comprising an active virtual machine identification input and a memory access location input [col. 27, lines 30 – 42], said active virtual machine identification input coupled [col. 27, lines 30 – 42] with said active virtual machine identification output [col. 27, line 60 – col. 28, line 3] and said memory access location input coupled with said memory access location output of said integrated circuit chip [PMU 1700, Fig. 17; col. 29, lines 28 – 39], said external memory protection component comprising a memory access error output [col. 31, lines 25 – 35], said memory access error output coupled with said memory access error input [col. 31, lines 35 – 40];

wherein said external memory protection component indicates a memory access error via said memory access error output when said memory access location input indicates memory location not associated with a virtual machine identified by said active virtual machine identification output [col. 31, lines 35 – 56].

14. As to claim 2, Gee teaches said running step comprises running at least two Java virtual machines [col. 23, line 65 – col. 24, line 18 and col. 23, lines 18 – 29].

15. As to claim 3, Gee teaches activating a specific independent virtual machine of said plurality of independent virtual machines [col. 25, lines 40 – 50].

16. As to claim 4, Gee teaches using a timer to define an activation period of an activated virtual machine [col. 28, lines 52 – 67].

17. As to claim 5, Gee teaches assigning a memory region to at least one independent virtual machine of the plurality of independent virtual machines [Preparation involves allocation of static storage and the creation of any data structures, such as method tables, that are used internally by the JVM; col. 6, lines 13 – 26].

18. As to claim 6, Gee teaches protecting a virtual machine's memory region from accesses by a different virtual machine [col. 27, lines 23 – 30].

19. As to claim 7, Gee teaches screening a memory access [col. 27, line 60 – col. 28, line 2]; and generating an abort interrupt signal to abort an access to a memory region of a nonactivated virtual machine [col. 28, line 3 – 13].

20. As to claim 8, Gee teaches outputting the identity of the activated virtual machine to a memory management component [active partition ID is maintained by a single partition register 1746; col. 30, lines 55 – 67].

21. As to claim 9, Gee teaches identifying, by the memory management component, the memory region assigned to the activated virtual machine [col. 30, lines 27 – 45].

22. As to claim 10, Gee teaches monitoring address lines to abort attempted memory accesses to a protected memory region [col. 27, line 60 – col. 28, line 2].

23. As to claim 11, Gee teaches aborting an attempted access to a protected memory region by generating an error signal [col. 31, lines 35 – 40].

24. As to claim 12, Gee teaches aborting an attempted access to a protected memory region by generating a prioritized nonmaskable interrupt signal [col. 28, lines 45 – 51].

25. As to claim 13, Gee teaches aborting an attempted access to a protected memory region by generating a highest priority [col. 21, lines 6 – 8] prioritized nonmaskable interrupt signal [col. 28, lines 45 – 51].

26. As to claim 14, Gee teaches receiving a maskable interrupt signal [col. 28, lines 3 – 15].

27. As to claim 15, Gee teaches latching a received maskable interrupt signal [col. 31, lines 57 – 64].

28. As to claim 16, Gee teaches latching a received maskable interrupt signal into a virtual interrupt latch component even though the independent virtual machine with which it is associated is not the activated independent virtual machine at the time the received maskable interrupt signal is received [col. 31, line 57 – col. 32, line 6].

29. As to claim 17, Gee teaches transferring the maskable interrupt signal, upon activation of its associated virtual machine, from the virtual interrupt latch component to a global interrupt mask register [col. 32, lines 7 – 31].

30. As to claim 18, Gee teaches transferring the maskable interrupt signal, upon activation of its associated virtual machine, from the virtual interrupt latch component [col. 32, lines 7 – 31] to a local mask register [col. 15, lines 33 – 37].

31. As to claim 19, Gee teaches sending the maskable interrupt signal, upon activation of its associated virtual machine [col. 32, lines 7 – 31], to a priority encoder after said steps of transferring and communicating [col. 10, lines 40 – 50].

32. As to claim 20, Gee teaches holding the received maskable interrupt signal in the virtual interrupt latch component until the independent virtual machine with which it is

associated has been activated [col. 31, line 57 – col. 32, line 6]; and servicing the received maskable interrupt signal during the time period that its associated independent virtual machine has been activated [col. 32, lines 7 – 31].

33. As to claim 21, Gee teaches discerning whether the independent virtual machine associated with the received maskable interrupt signal is the activated independent virtual machine; and ignoring the received maskable interrupt signal if it is discerned that the independent virtual machine with which the received maskable interrupt signal is associated is not the currently activated independent virtual machine [col. 28, lines 52 – 67].

34. As to claim 22, Gee teaches receiving a nonmaskable interrupt signal [col. 25, lines 19 – 29].

35. As to claim 23, Gee teaches receiving a nonmaskable interrupt signal indicating a power supply interruption [col. 26, lines 18 – 30].

36. As to claim 24, Gee teaches receiving a nonmaskable interrupt signal indicating activation of a different independent virtual machine [col. 28, lines 45 – 52].

37. As to claim 25, Gee teaches receiving a nonmaskable interrupt signal indicating an application specific event [col. 23, lines 30 – 43 and col. 24, liens 54 – 65].

38. As to claim 26, Gee teaches receiving a nonmaskable interrupt signal indicating a prohibited memory access attempt [col. 28, lines 3 – 13].

39. As to claim 27, Gee teaches reserving the highest priority for interrupt signals indicating a prohibited memory access attempt [col. 21, lines 6 – 8]; and wherein said receiving step comprises receiving a nonmaskable interrupt signal indicating a prohibited memory access attempt [col. 28, lines 3 – 13].

40. As to claim 28, Gee teaches suspending execution of the activated independent virtual machine upon receipt of a nonmaskable interrupt signal indicating a prohibited memory access attempt [col. 31, lines 44 – 49].

41. As to claim 30, Gee teaches interrupt controller component further comprises a plurality of virtual interrupt latch components [col. 10, lines 43 – 57].

42. As to claim 31, Gee teaches a plurality of global interrupt mask registers [a mask to the interrupt controller; col. 10, line 57 – col. 11, lines 9].

43. As to claim 32, Gee teaches a plurality of global interrupt mask registers, and wherein each global interrupt mask register is coupled with one of the virtual interrupt latch components [col. 20, line – col. 21, line 5].

44. As to claims 33 and 36, Gee teaches a local mask register coupled with said plurality of global interrupt mask registers [col. 15, lines 33 – 37].

45. As to claims 34 and 37, Gee teaches a priority encoder coupled with said local mask register [col. 10, lines 43 – 50].

46. **Claim 35 is rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent No. 6,728,746 to Murase et al. [hereinafter Murase].**

47. As to claim 35, Murase as modified teaches an interrupt controller [col. 18, lines 38 – 56] for a multiple virtual machine environment [col. 13, lines 1 – 11], comprising:

an interrupt signal input [AVM 210 causes an interruption in the logic machine in the machine 200 corresponding to the relative machine; col. 20, lines 25 – 28];

a plurality of virtual interrupt latch components [n logic machines 220-1, . . . 220-n; col. 13, lines 1 – 11] coupled with said interrupt signal input; and

a plurality of global interrupt mask registers [shared memory 100; col. 13, lines 32 – 43];

wherein each global interrupt mask register of said plurality of global interrupt mask registers is coupled with one of the virtual interrupt latch components [col. 13, line 55 – col. 14, line 5].

Claim Rejections - 35 USC § 103

48. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

49. **Claims 1, 29 and 38 are rejected under 35 U.S.C. 103(a) as being unpatentable over Murase in view of U.S. Patent No. 5,721,922 to Dingwall.**

50. As to claim 1, Murase teaches the invention substantially as claimed including a method for managing interrupts [col. 23, line 52 – col. 24, line 55] in a multiple virtual machine environment [col. 13, lines 1 – 11], comprising the steps of:

running concurrently a plurality of independent virtual machines [virtual machines 200 and 300, Fig. 12; col. 13, lines 1 – 12], each virtual machine having associated therewith a plurality of anticipated interrupt signal types [causes an interruption in the inter-machine communication control part 321 in the logic machine; col. 19, lines 47 – 57];

receiving a plurality of interrupt signals [AVM 210 causes an interruption in the logic machine in the machine 200 corresponding to the relative machine; col. 20, lines 25 – 28]; and

servicing the interrupt signal [machine controller 340 of the machine 300 honors the communication request and confirms that the communication has as its destination the machine 300 by referring to the real machine; col. 20, lines 34 – 45].

51. Although Murase teaches the invention substantially as claimed, Murase does not specifically teach interrupt priority and servicing the interrupt with the highest priority first.

However, Dingwall teaches a plurality of virtual machines [a set of DOS virtual machines 24, Fig. 2; col. 2, lines 37 – 48] determining which interrupt signal of the plurality of received interrupt signals has the highest priority and servicing the interrupt signal determined to have the highest priority [real-time scheduler 30 allows interrupt handlers 32 to make real-time tasks 34 ready for execution without preemption occurring...the real-time scheduler 30 allows the scheduling lock to be released causing any high priority, ready real-time tasks to preempt the current process; col. 3, lines 13 – 32].

52. It would have been obvious to a person of ordinary skill in the art at the time of the invention to apply the teaching of determining interrupt signal priority and servicing the signal with the highest priority as taught by Dingwall to the invention of Murase because this prevent interrupt latency and allows for real-time processing [col. 1, lines 42 – 44 and lines 59 – 60 of Dingwall].

53. As to claim 29, Murase as modified teaches an interrupt management system [col. 23, line 52 – col. 24, line 55 of Murase] for an apparatus capable of running multiple concurrent virtual machines [col. 13, lines 1 – 11 of Murase], comprising:

a timer component comprising a plurality of virtual machine timers [machine 200 effects monitoring by a timer; col. 31, lines 41 – 54 of Murase], said timer component

further comprising an active virtual machine switch signal output [col. 29, lines 52 – 60 of Murase];

a multiple virtual machine control component [col. 13, lines 1 – 11 of Murase], comprising an active virtual machine identification signal output [identification of a call originating logic machine and a call receiving logic machine; col. 16, lines 46 – 56 of Murase];

a processor component [service processors 290 and 390, Fig. 50; col. 33, lines 9 – 28 of Murase], coupled with said timer component;

an interrupt controller component [machine controller of Murase] coupled with said processor component [col. 33, lines 9 – 28 of Murase] and with said timer component [col. 31, lines 41 – 54 of Murase], said interrupt controller component comprising an active virtual machine identification signal input [310 logic machine identifiers of the logic machines; col. 18, lines 38 – 56 of Murase] coupled with said active virtual machine identification signal output, said interrupt controller component also comprising an interrupt signal input [machine controller 340 inquires of the AVM 310 logic machine identifiers of the logic machines having a logic path connected to the AVM 310 and the operating state of the logic machines; col. 18, lines 38 – 56 of Murase]; and

a memory component storing interrupt handler code [application specific interrupt handlers 32; col. 3, lines 6 – 13 of Dingwall].

54. As to claim 38, Murase as modified teaches a processor-based interrupt signal management system [col. 23, line 52 – col. 24, line 55 of Murase] for a multiple virtual machine environment [col. 13, lines 1 – 11 of Murase], comprising:

an integrated circuit chip [col. 33, lines 9 – 28 of Murase], comprising;

a processor component [service processors 290 and 390, Fig. 50; col. 33, lines 9 – 28 of Murase];

a multiple virtual machine management component [col. 23, line 52 – col. 24, line 55 of Murase] coupled with said processor component, said multiple virtual machine management component comprising a plurality of virtual machine [col. 13, lines 1 – 11 of Murase] activation timer components [col. 31, lines 41 – 54 of Murase];

a memory component coupled with said processor component, said memory component comprising interrupt handler code [col. 3, lines 6 – 13 of Dingwall];

a memory access error input [col. 31, lines 40 – 44 of Murase];

an active virtual machine identification output [col. 16, lines 46 – 56 of Murase]; and

a memory access location output [col. 15, lines 13 – 22 of Murase]; and

an external memory protection component, not located on said integrated circuit chip, comprising an active virtual machine identification input [col. 18, lines 38 – 56 of Murase] and a memory access location input [col. 15, lines 13 – 22 of Murase], said active virtual machine identification input coupled with said active virtual machine identification output and said memory access location input coupled with said memory access location output of said integrated circuit chip [machine controller 340 inquires of

the AVM 310 logic machine identifiers of the logic machines having a logic path connected to the AVM 310 and the operating state of the logic machines; col. 18, lines 38 – 56 of Murase], said external memory protection component comprising a memory access error output, said memory access error output coupled with said memory access error input [col. 31, lines 40 – 44 of Murase];

wherein said external memory protection component indicates a memory access error via said memory access error output when said memory access location input indicates memory location not associated with a virtual machine identified by said active virtual machine identification output [col. 33, lines 9 – 28 of Murase].

Conclusion

55. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

U.S. Patent No. 6,223,202 to Bayeh teaches a system for enabling multiple virtual machines to execute on a single server, using virtual machine pooling.

56. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Li B. Zhen whose telephone number is (571) 272-3768. The examiner can normally be reached on Mon - Fri, 8:30am - 5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Li B. Zhen
Examiner
Art Unit 2194

lbz

MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100